

## AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

### Listing of Claims:

- 1           1. (Currently amended) A method that predicts a result produced by a  
2 section of code in order to support speculative program execution, the section of  
3 code including a plurality of program instructions, the method comprising:  
4           executing the section of code within a program using a head thread,  
5 wherein executing the section of code produces the result;  
6           before the head thread produces the result, generating a predicted result to  
7 be used in place of the result;  
8           allowing a speculative thread to speculatively execute subsequent code  
9 within the program using the predicted result, wherein the subsequent code  
10 follows the section of code in an execution stream of the program, ~~and wherein~~  
11 ~~speculatively executing the subsequent code involves performing one of:~~  
12                     ~~a speculative method invocation to speculatively execute~~  
13                     ~~the subsequent code;~~  
14                     ~~a speculative function call to speculatively execute the~~  
15                     ~~subsequent code; and~~  
16                     ~~a speculative procedure call to speculatively execute the~~  
17                     ~~subsequent code;~~  
18           ~~wherein the head thread and all speculative threads execute instructions~~  
19 ~~from separate instruction caches; and~~

20           after the head thread finishes executing the section of code, determining if  
21   a difference between the predicted result and the result generated by the head  
22   thread affected execution of the speculative thread;  
23           if the difference affected execution of the speculative thread, executing the  
24   subsequent code again using the result generated by the head thread; and  
25           if the difference did not affect execution of the speculative thread,  
26   performing a join operation to merge state associated with the speculative thread  
27   with state associated with the head thread;  
28           wherein during a write operation to a memory element by the head thread,  
29   the write operation involves:  
30                   performing the write operation to a primary version of the  
31                   memory element,  
32                   checking status information associated with the memory  
33                   element to determine if the memory element has been read by the  
34                   speculative thread,  
35                   if the memory element has been read by the speculative  
36                   thread, causing the speculative thread to roll back so that the  
37                   speculative thread can read a result of the write operation, and  
38                   if the memory element has not been read by the speculative  
39                   thread, performing the write operation to a space-time dimensioned  
40                   version of the memory element if the space-time dimensioned  
41                   version exists; and  
42           wherein performing the join operation involves merging the space-time  
43   dimensioned version of the memory element into the primary version of the  
44   memory element and discarding the space-time dimensioned version of the  
45   memory element.

1           2. (Original) The method of claim 1, wherein executing the subsequent  
2 code again involves performing a rollback operation for the speculative thread to  
3 undo actions performed by the speculative thread.

1           3. (Original) The method of claim 1, wherein determining if the difference  
2 affected execution of the speculative thread involves determining if the  
3 speculative thread accessed the predicted result.

1           4. (Original) The method of claim 1, wherein determining if the difference  
2 affected execution of the speculative thread involves determining if the predicted  
3 result differs from the result generated by the head thread.

1           5. (Original) The method of claim 1, wherein generating the predicted  
2 result involves looking up a value based upon a program counter for the program.

1           6. (Original) The method of claim 5, wherein generating the predicted  
2 result involves additionally looking up the value based upon at least one  
3 previously generated value for the result.

1           7. (Original) The method of claim 5, wherein generating the predicted  
2 result involves performing a function on the value.

1           8. (Original) The method of claim 1, wherein executing the section of code  
2 involves performing one of:  
3           a method invocation to execute the section of code;  
4           a function call to execute the section of code; and  
5           a procedure call to execute the section of code.

1           9. (Original) The method of claim 1, wherein the section of code is a body  
2 of a loop in the program, and the result is a loop carried dependency for the loop.

1           10-11 (Canceled).

1           12. (Currently amended) An apparatus that facilitates predicting a result  
2 produced by a section of code in order to support speculative program execution,  
3 the section of code including a plurality of program instructions, the apparatus  
4 comprising:

5           a head thread that is configured to execute the section of code within a  
6 program, wherein executing the section of code produces the result;

7           a prediction mechanism that is configured to generate a predicted result to  
8 be used in place of the result before the head thread produces the result;

9           a speculative thread that is configured to speculatively execute subsequent  
10 code within the program using the predicted result, wherein the subsequent code  
11 follows the section of code in an execution stream of the program, and wherein  
12 ~~speculatively executing the subsequent code involves performing one of:~~

13                       ~~a speculative method invocation to speculatively execute~~  
14                       ~~the subsequent code;~~

15                       ~~a speculative function call to speculatively execute the~~  
16                       ~~subsequent code; and~~

17                       ~~a speculative procedure call to speculatively execute the~~  
18                       ~~subsequent code;~~

19           ~~wherein the head thread and all speculative threads execute instructions~~  
20 ~~from separate instruction caches; and~~

21           a determination mechanism that is configured to determine if a difference  
22 between the predicted result and the result generated by the head thread affected  
23 execution of the speculative thread; and

24 a joining mechanism that is configured to merge state associated with the  
25 speculative thread with state associated with the head thread if the difference did  
26 not affect execution of the speculative thread, wherein the joining mechanism is  
27 configured to:  
28 merge the space-time dimensioned version of the memory  
29 element into the primary version of the memory element, and to  
30 discard the space-time dimensioned version of the memory  
31 element; and  
32 a mechanism that performs write operations for the head thread, the  
33 mechanism being configured to:  
34 perform a write operation to a primary version of a memory  
35 element,  
36 check status information associated with the memory  
37 element to determine if the memory element has been read by the  
38 speculative thread,  
39 cause the speculative thread to roll back so that the  
40 speculative thread can read a result of the write operation if the  
41 memory element has been read by the speculative thread, and  
42 perform the write operation to a space-time dimensioned  
43 version of the memory element if the space-time dimensioned  
44 version exists and if the memory element has not been read by the  
45 speculative thread;  
46 wherein if the difference affected execution of the speculative thread, the  
47 apparatus is configured to execute the subsequent code again using the result  
48 generated by the head thread.

1           13. (Original) The apparatus of claim 12, wherein while executing the  
2 subsequent code again, the apparatus is configured to perform a rollback operation  
3 for the speculative thread to undo actions performed by the speculative thread.

1           14. (Original) The apparatus of claim 12, wherein the determination  
2 mechanism is configured to determine if the speculative thread accessed the  
3 predicted result.

1           15. (Original) The apparatus of claim 12, wherein the determination  
2 mechanism is configured to determine if the predicted result differs from the  
3 result generated by the head thread.

1           16. (Original) The apparatus of claim 12, wherein the prediction  
2 mechanism is configured to generate the predicted result by looking up a value  
3 based upon a program counter for the program.

1           17. (Original) The apparatus of claim 16, wherein the prediction  
2 mechanism is configured to generate the predicted result by additionally looking  
3 up the value based upon at least one previously generated value for the result.

1           18. (Original) The apparatus of claim 16, wherein the prediction  
2 mechanism is configured to generate the predicted result by performing a function  
3 on the value.

1           19. (Original) The apparatus of claim 12, wherein the section of code  
2 includes one of, a method, a function and a procedure.

1           20. (Original) The apparatus of claim 12, wherein the section of code is a  
2 body of a loop in the program, and the result is a loop carried dependency for the  
3 loop.

1           21-22 (Canceled).

1           23. (Currently amended) A computer-readable storage medium storing  
2 instructions that when executed by a computer cause the computer to perform a  
3 method that predicts a result produced by a section of code in order to support  
4 speculative program execution, the section of code including a plurality of  
5 program instructions, the method comprising:

6           executing the section of code within a program using a head thread,  
7 wherein executing the section of code produces the result;  
8           before the head thread produces the result, generating a predicted result to  
9 be used in place of the result;

10          allowing a speculative thread to speculatively execute subsequent code  
11 within the program using the predicted result, wherein the subsequent code  
12 follows the section of code in an execution stream of the program, and wherein  
13 speculatively executing the subsequent code involves performing one of:

14                   a speculative method invocation to speculatively execute  
15                   the subsequent code;  
16                   a speculative function call to speculatively execute the  
17                   subsequent code; and  
18                   a speculative procedure call to speculatively execute the  
19                   subsequent code;

20          wherein the head thread and all speculative threads execute instructions  
21 from separate instruction caches; and

22           after the head thread finishes executing the section of code, determining if  
23   a difference between the predicted result and the result generated by the head  
24   thread affected execution of the speculative thread;  
25           if the difference affected execution of the speculative thread, executing the  
26   subsequent code again using the result generated by the head thread; and  
27           if the difference did not affect execution of the speculative thread,  
28   performing a join operation to merge state associated with the speculative thread  
29   with state associated with the head thread;  
30           wherein during a write operation to a memory element by the head thread,  
31   the write operation involves:  
32                   performing the write operation to a primary version of the  
33                   memory element,  
34                   checking status information associated with the memory  
35                   element to determine if the memory element has been read by the  
36                   speculative thread,  
37                   if the memory element has been read by the speculative  
38                   thread, causing the speculative thread to roll back so that the  
39                   speculative thread can read a result of the write operation, and  
40                   if the memory element has not been read by the speculative  
41                   thread, performing the write operation to a space-time dimensioned  
42                   version of the memory element if the space-time dimensioned  
43                   version exists; and  
44           wherein performing the join operation involves merging the space-time  
45   dimensioned version of the memory element into the primary version of the  
46   memory element and discarding the space-time dimensioned version of the  
47   memory element.



1           24. (Original) The computer-readable storage medium of claim 23,  
2 wherein executing the subsequent code again involves performing a rollback  
3 operation for the speculative thread to undo actions performed by the speculative  
4 thread.